22.2 124MS/s Pixel-Pipelined Motion-JPEG 2000 Codec without Tile Memory

Yu-Wei Chang, Hung-Chi Fang, Chih-Chi Cheng, Chun-Chia Chen, Chung-Jr Lian, Shao-Yi Chien, Liang-Gee Chen

National Taiwan University, Taipei, Taiwan

JPEG 2000 uses two key components, the discrete wavelet transform (DWT) and embedded block coding with optimized truncation (EBCOT), to achieve excellent coding efficiency and numerous features, such as scalability and region of interest. The scalability comes from the multi-level decompositions of the DWT and the embedded block coding (EBC) of the EBCOT.

There are three challenges in the design of an efficient JPEG 2000 codec for HD video. Firstly, the large data rate between the DWT and the EBCOT requires either large on-chip SRAM or high SDRAM bandwidth. Secondly, complicated control and irregular dataflow of the DWT and the EBC require large implementation area to meet the high throughput requirements. Thirdly, hardware sharing between the encoder and the decoder is difficult due to different computational characteristics and dataflow.

This paper presents efficient techniques to overcome the above challenges. Figure 22.2.1 shows the features of the 1920×1080 motion-JPEG 2000 codec chip we have developed. The core size is 20.1mm^2 with 0.18μm CMOS technology. It contains 1155k logic gates and 19.9kB of SRAM. The power consumption is 385mW at 1.8V and 42MHz.

Figure 22.2.2 shows the block diagram of the codec chip. It contains a Main Controller, a 3-level DWT module, three embedded block coding (EBC) modules, a rate-distortion optimization (RDO) controller, and a bit stream controller (BSC). The RDO controller maximizes image quality at a given target bit rate. Both the DWT and the EBC are pixel-pipelined so that no tile memory is required between the DWT and the EBC. Moreover, both the encoding and the decoding are one-pass, that is, no coefficient transmission to or from the SDRAM is required.

There are two problems that make a pixel-pipelined architecture challenging. Firstly, the dataflow patterns of the DWT and the EBC are quite different; the DWT generates the coefficients in a subband-interleaving manner while the EBC encodes a code-block within one subband at a time. Secondly, the DWT is a word-level algorithm while the EBC is a bit-level one. Therefore, a tile-level pipeline schedule is used in all previous known works. This introduces either high bandwidth for those storing tiles in the SDRAM [1] or high cost for those storing in on-chip SRAM [2]. In [1], the bandwidth requirement is so high that two buses are needed, and each operates at twice the frequency of the core. In [2], a 192kB SRAM is required for a 256×256 tile, which would occupy half of the silicon area.

A level-switched schedule is proposed to solve the above two problems. It eliminates the 192kB on-chip SRAM for the architecture in [2] and reduces the 310MB/s SDRAM bandwidth for the architecture in [1]. The detailed encoding schedule for both the DWT and the EBC is shown in Fig. 22.2.3. Each computation state requires 256 cycles for encoding either one stripe of a 64×64 code-block or two stripes of a 32×32 code-block. The tile memory is eliminated by encoding the stripes in multiple subbands/levels in an interleaved manner. Three EBC modules match the output dataflow of the DWT. With a pixel-pipelined dataflow, the EBC is capable of processing one coefficient per cycle so that a generated DWT coefficient is encoded immediately. Therefore, neither tile memory nor internal buffer between the DWT and the EBC is required.

To enable this schedule, a level-switch DWT as shown in Fig. 22.2.4 is proposed. At each computation state, the DWT decomposes a 128×8 or 64×16 block in a tile and generates 64×4 or 32×8 coefficients, respectively, in each subband. The DWT switches to the next level decomposition as soon as the data in the LL Band Buffer is enough for a computation state. Therefore, the buffer size is minimized to 5.2kB. To match the level-switched schedule of the DWT, an additional 2.5kB buffer is required for each EBC module to store the coding states of the unfinished code-blocks. The decoding schedule is opposite to the encoding schedule. Thus, the memory requirement for decoding equals that for encoding. By use of this schedule, the 192kB tile memory is eliminated with an additional 12.7kB on-chip SRAM.

Two hardware sharing techniques are developed to design the codec with a unified hardware. Firstly, the schedules for encoding and decoding are matched to achieve 100% memory sharing for the DWT and the EBC. Secondly, the reconfigurable filter core and arithmetic coder for the DWT and the EBC save 489k logic gates. By use of these techniques, the logic gates of the codec are 136% (118%) larger than those of the encoder (decoder). The silicon area is reduced by 40%.

The EBC is the design bottleneck of a high-throughput JPEG 2000 codec. In [2], a word-level EBC encoder is used to increase the throughput. However, the throughput depends on the complexity of image source and the target image quality. In this work, a word-level EBC codec, which guarantees one coefficient encoding/decoding per cycle, is developed. Figure 22.2.5 shows the block diagram of the EBC codec, which processes a 10b DWT coefficient per cycle. The four-symbol arithmetic coder (FAC) is proposed to encode/decode all the contexts from a bit plane in one cycle. The State Memory buffers the coding states in the state register bank (SRB) when switching to another code-block, and loads the states back to the SRB before continuing the unfinished code-block.

Figure 22.2.6 shows the effectiveness of the proposed techniques on the reduction of memory requirements. Although the SDRAM bandwidth is reduced to 37% by embedding the tile memory, the on-chip memory is too large, dramatically increasing the silicon cost. However, with the proposed techniques, the on-chip tile memory is eliminated while the bandwidth is kept the same.

A performance index (PI), defined as throughput per unit area at 1MHz, is used to make a fair comparison of the existing works. The PI of this chip is 0.148 ($^{124}/_{20\times42}$). The estimated area of the JPEG 2000 encoder/decoder core in [1] is 13 and 6.5mm^2, respectively. Therefore, the PI for the encoder/decoder is 0.100/0.100 ($^{70}/_{13\times54}$ / $^{35}/_{6.5\times54}$). Hence, this codec is 1.48 times better than both the encoder and decoder in [1]. Moreover, the SDRAM bandwidth of this chip is 280MB/s less than that of [1]. The improvement mainly comes from the proposed level-switched schedule.

Figure 22.2.7 is the micrograph of the chip. Both encoding and decoding functions achieve 124MS/s data rate. The memory sharing and the reconfigurable processing elements reduce the silicon area. The level-switched schedule eliminates tile memory by matching the dataflow and the throughput of the DWT and the EBC.

Acknowledgements:
The authors thank Prof. Homer Chen, Prof. Chien-Mo Li, and members of DSP/IC Design Lab for contributions and discussions. The authors also acknowledge Chip Implementation Center for supporting fabrication.

References:
[1] H. Yamauchi, et. al., "1440×1080 Pixel, 30 Frames per Second Motion-JPEG 2000 Codec for HD-Movie Transmission," *IEEE J. Solid-State Circuits,* pp. 331-341, Jan., 2005.
[2] H.-C. Fang, et. al., "81 MS/s JPEG 2000 Single-Chip Encoder with Rate-Distortion Optimization," *ISSCC Dig. Tech. Papers*, Feb., 2004.

Technology	TSMC 0.18μm 1P6M CMOS
Supply Voltage	1.8V
Core Area	5.37x3.75mm^2
Logic Gates	1155K (2-input NAND gate)
SRAM	19.9kB
Operating Frequency	42MHz
Power	385mW for lossless encoding 379mW for lossless decoding
Image Size	Up to 64K x 64K
Processing Rate	124M Sample/sec 1920x1080 30Frames/s 4:2:2 (support lossless and lossy coding)
DWT	(5,3), (9,7) filter, 3-level
Tile Size	256x256
Code-block Size	64x64

Figure 22.2.1: Chip features.

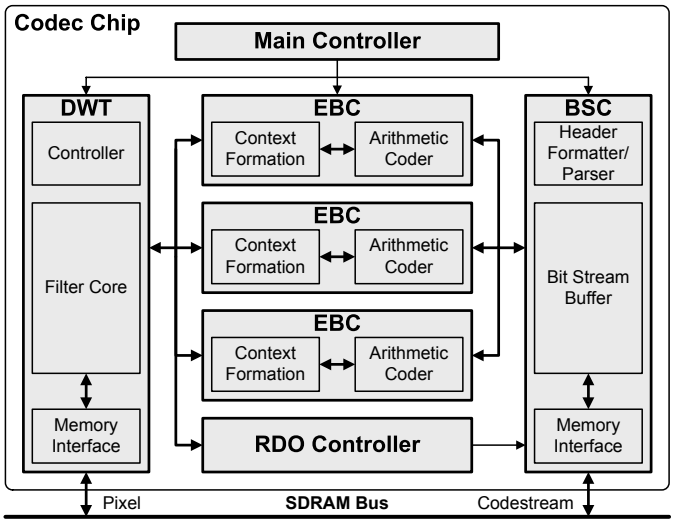

Figure 22.2.2: Block diagram of the JPEG 2000 codec system.

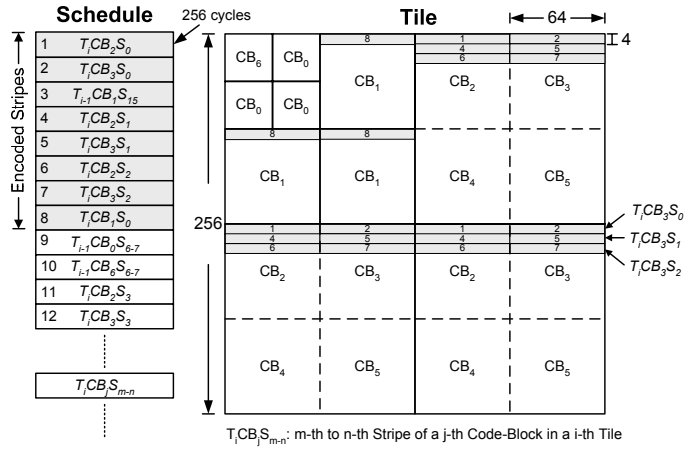

Figure 22.2.3: Level-switched schedule for JPEG 2000 encoder.

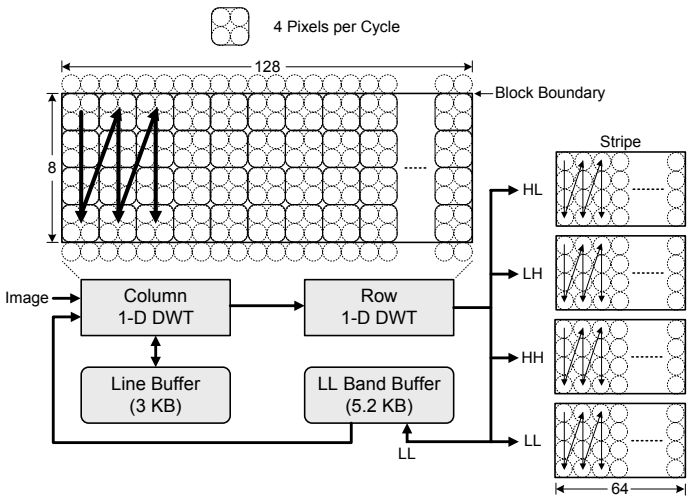

Figure 22.2.4: Dataflow of the DWT for forward transformation.

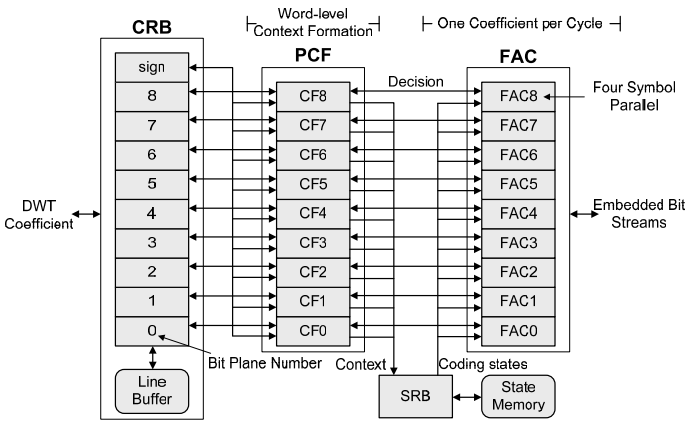

Figure 22.2.5: Word-level EBC codec architecture.

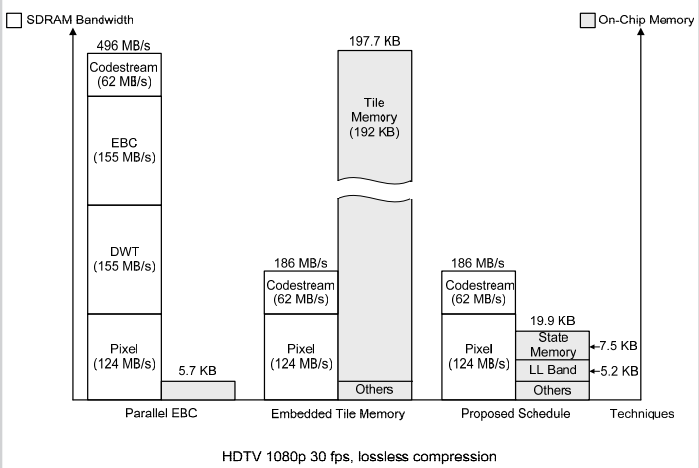

Figure 22.2.6: Effectiveness of SDRAM bandwidth and on-chip memory size reduction.

Continued on Page 661

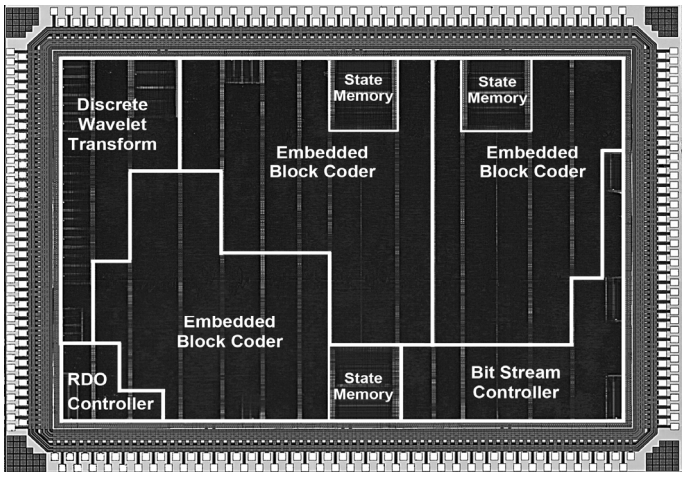

Figure 22.2.7: Die micrograph of the JPEG 2000 Codec.

